Universal dependencies for Turkish

Sulubacak, Umut

The Association for Computational Linguistics
2016-12

þÿ S u l u b a c a k , U , G ö k 1 r m a k , M , T y e r s , F , Ç ö l t e k i n , Ç , N i v r e , J & E
Universal dependencies for Turkish . in Y Matsumoto & R Prasad (eds) , Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers . The Association for Computational Linguistics , Osaka, Japan , pp. 3444-3454 , International Conference on Computational Linguistics , Osaka , Japan , 11/12/2016 .

http://hdl.handle.net/10138/237163

Universal Dependencies for Turkish

Umut Sulubacak ⋆ and **Memduh Gökırmak** †
Department of Computer Engineering
Istanbul Technical University
34469 Istanbul, Turkey
{⋆sulubacak|†gokirmak}@itu.edu.tr

Francis M. Tyers
HSL-fakultehta
UiT Norgga árktalaš universitehta
N-9018 Tromsø, Norway
francis.tyers@uit.no

Çağrı Çöltekin
Department of Linguistics
University of Tübingen
72074 Tübingen, Germany
ccoltekin@sfs.uni-tuebingen.de

Joakim Nivre
Department of Linguistics and Philology
Uppsala University
75126 Uppsala, Sweden
joakim.nivre@lingfil.uu.se

Gülşen Eryiğit
Department of Computer Engineering
Istanbul Technical University
34469 Istanbul, Turkey
gulsen.cebiroglu@itu.edu.tr

Abstract

The Universal Dependencies (UD) project was conceived after the substantial recent interest in unifying annotation schemes across languages. With its own annotation principles and abstract inventory for parts of speech, morphosyntactic features and dependency relations, UD aims to facilitate multilingual parser development, cross-lingual learning, and parsing research from a language typology perspective. This paper presents the Turkish IMST-UD Treebank, the first Turkish treebank to be in a UD release. The IMST-UD Treebank was automatically converted from the IMST Treebank, which was also recently released. We describe this conversion procedure in detail, complete with mapping tables. We also present our evaluation of the parsing performances of both versions of the IMST Treebank. Our findings suggest that the UD framework is at least as viable for Turkish as the original annotation framework of the IMST Treebank.

1 Introduction

The Universal Dependencies (UD)[1] project is an international collaborative project to make cross-linguistically consistent treebanks available for a wide variety of languages. Currently in version 1.3, the UD project covers 40 languages, including two Turkic languages: Kazakh, which was annotated from scratch, and Turkish, the creation of which is described in this paper.

The universal annotation guidelines of UD are based on the Google Universal Part-of-Speech Tagset (Petrov et al., 2012) for parts of speech, the Interset framework (Zeman, 2008) for morphological features, and Stanford Dependencies (De Marneffe et al., 2006; Tsarfaty, 2013; De Marneffe et al., 2014) for dependency relations. The objective of harmonizing annotation guidelines as far as possible is to make comparison of parsing results and investigating cross-linguistic methods across languages easier. This is achieved by a number of principles, including the primacy of content words, distinguishing core arguments from modifiers and distinguishing clausal constituents from nominals.

The IMST-UD Treebank was first released in UD version 1.3 and became the first Turkish treebank to be included in a UD release. The treebank was created by automatic conversion of the IMST

[1] http://universaldependencies.org/

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 3444–3454, Osaka, Japan, December 11-17 2016.

Treebank (Sulubacak et al., 2016), which is itself a reannotation of the METU-Sabancı Turkish Treebank (Oflazer et al., 2003; Atalay et al., 2003). Although the annotation framework of the IMST Treebank was revised, it is still fundamentally similar to that of the METU-Sabancı Treebank and radically different from the UD framework in both morphology and syntax.

In this paper, we describe the procedures employed in converting the annotation schemes of the IMST Treebank to the corresponding UD-compliant schemes. We also provide comparative statistics on the composition of the IMST Treebank before and after the conversion. Afterwards, we report our initial parsing results on the new IMST-UD Treebank in comparison with the original IMST Treebank. The paper is structured as follows: Section 2 discusses the conversion procedure, Section 3 describes the IMST Treebank and the relevant statistics, Section 4 explains the parsing tests and their analysis, and finally, Section 5 presents the conclusion.

2 Mapping

In this section, we describe the procedure we employed in mapping the original IMST Treebank to a UD-compliant framework. The UD-compliant grammatical representations to which we mapped the original annotation schemes were largely adapted from previous work in the subject (Çöltekin, 2015; Çöltekin, 2016). The original treebank was available in the CoNLL-X data format (Buchholz and Marsi, 2006), where sentences are bounded by empty lines, and every word has a separate row, each containing a tab-delimited array of morphosyntactic data pertaining to the word. In compliance with the UD standard, the converted sentences were output in the CoNLL-U format.[2]

The sections to follow present explanations and discussions on the procedures of mapping morphological and syntactic data, as well as some idiosyncratic linguistic phenomena. Quick reference tables were also provided where applicable, showing what conditions on the *source* unit are required to assign which properties to the *target* unit.

2.1 Segmentation

The inflectional group (IG) formalism (Oflazer, 1999; Hakkani-Tür et al., 2002) was designed to make the highly agglutinative typology of Turkish tractable for language processing. Since then, it has seen usage in many influential works (Oflazer, 2003; Eryiğit and Oflazer, 2006) and has become the de facto standard in parsing Turkish. According to the formalism, orthographic tokens are divided into morphosyntactic words from *derivational boundaries*.[3] These units are called the inflectional groups (IGs) of the token. The IG formalism establishes these, rather than orthographic tokens, as the syntactic units of the sentence.

The original IMST treebank also follows its predecessors in using the IG formalism. The rightmost IG governs the word, while every other IG depends on the next one in line with the exclusive relation DERIV. Though a computationally effective representation, IGs are in contradiction with the UD principles. The representation dictates that the rightmost IG (which is, more often than not, a function word) be the head, whereas the leftmost IG (which is always a content word) is made to be the deepest dependent. As this does not comply with the principle of the primacy of content words, IGs have been removed during the conversion to UD. As a substitute, some derivational morphemes were treated as unbound enclitics, segmented off of their host words, assigned parts of speech such as ADP and AUX, and made to depend on their stems. Other morphemes were merged with their stems and were either fully lexicalized or marked for complex morphology. By a lexicalized derivation we mean tokens for which the grammatical process of derivation is not represented, and the result of the derivation is considered to be the lemma. An example for this is shown with *küreselleşme* in Figure 1.

Table 1a outlines the derivations that were segmented off of their stems. The surface forms for each such segment was constructed with the help of a morphological synthesizer, by 1) compiling the morphological analysis of the whole token, then 2) removing the part that corresponds to the derivation and any

[2]The CoNLL-U format is itself a revised version of the CoNLL-X data format. A description of the format is maintained (at the time of writing) on the official UD website (`http://universaldependencies.org/format.html`).

[3]In this context, a *derivational boundary* is the boundary between a POS-changing derivational suffix (or zero morpheme) and the stem that it is added to.

Source		*Target*			
CPOSTAG	POSTAG	LEMMA	FORM	UPOSTAG	DEPREL
ADJ	AGT	*ci*	**SYNTHESIZE**	ADP	CASE
ADJ	FITFOR	*lik*	**SYNTHESIZE**	ADP	CASE
ADJ | NOUN	REL	*ki*	**SYNTHESIZE**	ADP	CASE
ADJ	WITH	*li*	**SYNTHESIZE**	ADP	CASE
ADJ	WITHOUT	*siz*	**SYNTHESIZE**	ADP	CASE
ADVERB	LY	*ce*	**SYNTHESIZE**	ADP	CASE
ADVERB	SINCE	*dir*	**SYNTHESIZE**	ADP	CASE
NOUN	NESS	*lik*	**SYNTHESIZE**	ADP	CASE
VERB	ZERO	*i*	**SYNTHESIZE**	AUX	COP

(a)

CPOSTAG	POSTAG
ADJ	INBETWEEN
ADJ	JUSTLIKE
ADJ	RELATED
NOUN	AGT
NOUN	DIM
VERB	ACQUIRE
VERB	BECOME

(b)

Table 1: *(a)* Segmentation of copulas and other derivations, and *(b)* lexicalized derivations.

following inflection, and finally 3) synthesizing the new form from this partial analysis. The segments were also assigned the lemmas and parts of speech given in the LEMMA and UPOSTAG columns of the table, and made to depend on their stems with the relation specified in the DEPREL column.

The derivations given in Tables 1a, 1b and 3 are made via the addition of various derivational suffixes. Each of these suffixes has several allomorphs according to vowel harmony (e.g. the agent-deriving suffix may have the following 16 forms: *–cı, –ci, –cu, –cü, –çı, –çi, –çu, –çü, –ıcı, –ici, –ucu, –ücü, –yıcı, –yici, –yucu, –yücü*), and sometimes there is no overt suffix (as in the third person singular copula, which is a zero morpheme). Moreover, words are often further inflected after derivation, or may be multiply derived, and the analysis of these cascading and overlapping suffixes is an ambiguous and unreliable process. Therefore, instead of derivational morphemes, the minor part-of-speech tags assigned to each word (given in the POSTAG column) were used to identify derivations.

Table 1b lists the derivations that were not considered sufficiently productive and merged with their stems. Although these derivations have varying degrees of productivity, words derived by them are largely confined to a limited group of fairly common derivations. The fact that these words were more often than not lexicalized in the original treebank served as our justification for the lexicalization. The *lexicalized token* was made to inherit the surface form, lemma, and all morphological and syntactic data from the *derivation*, as well as its dependents, before replacing both the *stem* and the *derivation*.

Table 3 summarizes the participle (verbal adjective), transgressive (verbal adverb) and gerund (verbal noun) derivations in the same manner. In compliance with the UD standard of encoding verb forms, the *merged token* was made to inherit the lemma of the *stem*, as well as the surface form, the CASE, PERSON[PSOR], NUMBER[PSOR] and TENSE features, the head index, and the dependents of the

Source		*Target*	
CPOSTAG	POSTAG	UPOSTAG	FEATS
ADJ	NUM	NUM	—
ADJ	—	ADJ	—
ADVERB	—	ADV	—
DET	—	DET	—
DUP	—	X	ECHO=RDP
CONJ	—	CONJ	—
INTERJ	—	INTJ	—
NOUN	NUM	NUM	—
NOUN	PROP | ABR	PROPN	—
NOUN	—	NOUN	—
POSTP	NEG	VERB	—
POSTP	QUES	AUX	—
POSTP	—	ADP	—
PRON	DEMONS	PRON	PRONTYPE=DEM
PRON	PERS	PRON	PRONTYPE=PRS
PRON	QUANT	PRON	PRONTYPE=IND
PRON	REFLEX	PRON	REFLEX=YES
PRON	—	PRON	—
PUNC	—	PUNCT	—
VERB	ZERO	AUX	—
VERB	—	VERB	—

Table 2: Part-of-speech tag mapping.

Source		*Target*	
CPOSTAG	POSTAG	UPOSTAG	FEATS
ADVERB	ADAMANTLY	VERB	VERBFORM=TRANS
ADVERB	AFTERDOINGSO	VERB	VERBFORM=TRANS
ADVERB	ASIF	VERB	VERBFORM=TRANS
ADVERB	ASLONGAS	VERB	VERBFORM=TRANS
ADVERB	BYDOINGSO	VERB	VERBFORM=TRANS
ADVERB	SINCEDOINGSO	VERB	VERBFORM=TRANS
ADVERB	WHILE	VERB	VERBFORM=TRANS
ADVERB	WHEN	VERB	VERBFORM=TRANS
ADVERB	WITHOUTBEINGABLETOHAVEDONESO	VERB	MOOD=ABIL | NEGATIVE=NEG | VERBFORM=TRANS
ADVERB	WITHOUTHAVINGDONESO	VERB	NEGATIVE=NEG | VERBFORM=TRANS
ADJ	AORPART	VERB	TENSE=AOR | VERBFORM=PART
ADJ	NARRPART	VERB	ASPECT=PERF | TENSE=PAST | VERBFORM=PART
ADJ	PASTPART	VERB	TENSE=PAST | VERBFORM=PART
ADJ	PRESPART	VERB	TENSE=PRES | VERBFORM=PART
ADJ	FUTPART	VERB	TENSE=FUT | VERBFORM=PART
NOUN	INF1	VERB	VERBFORM=GER
NOUN	INF2	VERB	VERBFORM=GER
NOUN	INF3	VERB	VERBFORM=GER

Table 3: Merging of verbal derivations (transgressives, participles and gerunds).

derivation. The *merged token* was also assigned a VERBFORM feature as designated by the mapping, along with ASPECT, MOOD, TENSE and NEGATIVE features, before replacing the *stem* and the *derivation*.

In addition to the derivations discussed previously in this section, there were some zero derivations in the original treebank that were immediately derived into other parts of speech without any inflection inbetween, such as when adjectives were derived into zero nouns before copular (verbal) derivations. These intermediate derivations held no morphosyntactic information and were eliminated in conversion.

2.2 Part-of-Speech Tags

The mapping of the UD part-of-speech tags are displayed in Table 2. Most parts of speech were mapped in a straightforward, one-to-one fashion, with a small number of exceptions. In some cases, extra morphological features were used for an expressive conversion.

2.3 Morphological Features

Table 4 shows the mapping of the morphological features. Derivational information was mostly kept in the minor part of speech (POSTAG) field in the original IMST Treebank. These tags were retained in the XPOSTAG field in the CoNLL-U output after the conversion. Using either a directly corresponding UD feature or a combination of other UD features, we were able to represent most of the information kept in these fields.

The TENSE, ASPECT and MOOD features are closely related and often fused in Turkish. In some cases, a multiply derived token may have more than one value for one of these features. Moreover, although the UD guidelines enforce these features for finite verbs, they were occasionally omitted in the IMST Treebank so that they would defer to a neutral value. Whenever one of these features had more than one corresponding value, we concatenated these values with a hyphen delimiter, except for multiple occurrences of the same feature value, and the cases specified in Table 4. If one of these features had no directly corresponding value, we assigned the implied default value (TENSE=PRES, ASPECT=PERF, and MOOD=IND). For instance, the feature sequence HASTILY | PROG1 was converted to ASPECT=PROG-RAPID | MOOD=IND | TENSE=PRES.

Source	*Target*
FEATS	FEATS
A1SG	PERSON=1 | NUMBER=SING
A2SG	PERSON=2 | NUMBER=SING
A3SG	PERSON=3 | NUMBER=SING
A1PL	PERSON=1 | NUMBER=PLUR
A2PL	PERSON=2 | NUMBER=PLUR
A3PL	PERSON=3 | NUMBER=PLUR
PNON	—
P1SG	PERSON[PSOR]=1 | NUMBER[PSOR]=SING
P2SG	PERSON[PSOR]=2 | NUMBER[PSOR]=SING
P3SG	PERSON[PSOR]=3 | NUMBER[PSOR]=SING
P1PL	PERSON[PSOR]=1 | NUMBER[PSOR]=PLUR
P2PL	PERSON[PSOR]=2 | NUMBER[PSOR]=PLUR
P3PL	PERSON[PSOR]=3 | NUMBER[PSOR]=PLUR
ABL	CASE=ABL
ACC	CASE=ACC
DAT	CASE=DAT
EQU	CASE=EQU
GEN	CASE=GEN
LOC	CASE=LOC
INS	CASE=INS
NOM	CASE=NOM
CARD	NUMTYPE=CARD
DIST	NUMTYPE=DIST
ORD	NUMTYPE=ORD

Source	*Target*
FEATS	FEATS
AOR	TENSE=AOR
FUT	TENSE=FUT
PAST | PAST	TENSE=PQP | REGISTER=INF
PAST	TENSE=PAST
PRES	TENSE=PRES
NARR | PAST	TENSE=PQP
NARR | NARR	TENSE=PQP | EVIDENTIALITY=NFH
NARR	TENSE=PAST | EVIDENTIALITY=NFH
HASTILY	ASPECT=RAPID
PROG1	ASPECT=PROG | REGISTER=INF
PROG2	ASPECT=PROG | REGISTER=FORM
REPEAT	ASPECT=DUR
STAY	ASPECT=DUR-PERF
ABLE	MOOD=ABIL
ALMOST	MOOD=PRO
COND	MOOD=CND
COP	MOOD=GEN
DESR	MOOD=DES
IMP	MOOD=IMP
NECES	MOOD=NEC
OPT	MOOD=OPT
NEG	NEGATIVE=NEG
POS	NEGATIVE=POS
CAUS	VOICE=CAU
PASS	VOICE=PASS

Table 4: Morphological feature mapping.

2.4 Dependency Relations

The mapping rules used in converting dependency relations are outlined in Tables 5, 6, 7, and 8. The conditions for these mapping rules are considerably more complex than for the parts of speech and the morphological features. More often than not, besides the original dependency relations, additional morphosyntactic and lexical data must be considered for an accurate mapping. Furthermore, the entire analysis of a given dependent may sometimes not suffice, and further data pertaining to the head token that governs that dependent must be considered as well (as specified under columns with *(head)* labels).

Table 5 shows the mappings for dependency relations that are essentially types of modifiers and determiners. The mapping conditions are exactly as arranged on the table, except for the mapping to the ADVCL relation, where if the word had the feature VERBFORM=GER, it was also required to have an adpositional dependent with a CASE dependency. This means having a CASE dependent on a verbal head, which is incompatible with the UD guidelines for the moment. However, as this is an issue that will be discussed in the future, we decided to wait and see whether a change in the guidelines will be made. Table 6 displays the rules for dependencies that denote multiword expressions and other compounds. Multiword expressions (MWEs) were mapped to five different UD relations dependending on their context. The remaining MWEs were converted according to their syntactic role in the sentence. For both of the groups covered in Tables 5 and 6, certain cases were only distinguishable by their lemmas. These cases are given in additional rows below each table.

Tables 7 and 8 show the mappings for the remaining dependency relations. These tables also give exact mapping conditions, except for tokens with OBJECT dependencies (Table 7), which were still mapped to CCOMP dependencies without a VERBFORM=GER feature if they had a copular dependent with a COP dependency. Table 8 is reserved for dependency conversions whose head indices were adjusted

Source				*Target*
DEPREL *(dep)*	CPOSTAG *(dep)*	FEATS *(dep)*	FEATS *(head)*	DEPREL
INTENSIFIER	ADV	—	—	ADVMOD:EMPH
NOT INTENSIFIER	ADV	—	—	ADVMOD
MODIFIER	—	—	VERBFORM=PART	ACL
MODIFIER	NUM	NUMTYPE=ORD | DIST	—	AMOD
MODIFIER	VERB	VERBFORM=GER | TRANS	—	ADVCL
MODIFIER	NUM	NUMTYPE=CARD	—	NUMMOD
MODIFIER	NOUN | PRON | PROPN	—	—	NMOD
POSSESSOR	NOUN	CASE=ABL	**NO** PERSON[PSOR]	NMOD
POSSESSOR	NOUN	—	PERSON[PSOR]	NMOD:POSS
DEPREL *(dep)*	CPOSTAG *(dep)*	LEMMA *(dep)*	LEMMA *(head)*	DEPREL
MODIFIER	ADJ	**NOT** *(hangi | nasıl | ne | nere)*	—	AMOD
MODIFIER	ADJ	*(hangi | nasıl | ne | nere)*	—	DET
DETERMINER	—	*(her | hiçbir | ne)*	**NOT** *(şey | yer | zaman)*	DET

Table 5: Dependency mapping: Modifiers and determiners.

Source					*Target*
DEPREL *(dep)*	CPOSTAG *(dep)*	CPOSTAG *(head)*	FEATS *(dep)*	FEATS *(head)*	DEPREL
POSSESSOR	NOUN	NOUN	CASE=NOM	**NO** PERSON[PSOR]	COMPOUND
MWE | MODIFIER	NUM	NUM	—	—	COMPOUND
MWE	X	X	ECHO=RDP	ECHO=RDP	COMPOUND:REDUP
MWE	PROPN	PROPN	—	—	NAME
DEPREL *(dep)*	CPOSTAG *(dep)*	CPOSTAG *(head)*	LEMMA *(dep)*	LEMMA *(head)*	DEPREL
MWE | DETERMINER	—	—	*(her | hiçbir | ne)*	*(şey | yer | zaman)*	MWE
MWE | MODIFIER	—	VERB	—	*(bulun | et | ol | kıl)*	COMPOUND:LVC

Table 6: Dependency mapping: Multiword expressions and other compounds.

Source			*Target*
DEPREL	CPOSTAG	POSTAG	DEPREL
APPOSITION	NOUN	—	APPOS
APPOSITION	VERB	—	PARATAXIS
OBJECT	VERB	VERBFORM=GER	CCOMP
OBJECT	—	**NO** VERBFORM=GER	DOBJ
PREDICATE	—	—	ROOT
SUBJECT	VERB	VERBFORM=GER	CSUBJ
SUBJECT	—	**NO** VERBFORM=GER	NSUBJ

Table 7: Dependency mapping: Other dependencies, keeping the typology.

along with their dependency relations. For the mappings marked **SWAP** in the HEAD column, the direction of the dependency was also reversed. The original dependent became the new head and vice versa, and the dependents of these tokens were swapped. For those marked **CLAUSAL**, the head of the dependency (usually the sentence root in the original IMST Treebank) was updated to the head of the clause in which the token occurs. If no such clause exists, the head of the sentence was assigned instead.

Source				*Target*	
DEPREL *(dep)*	CPOSTAG *(dep)*	POSTAG *(dep)*	LEMMA *(dep)*	DEPREL	HEAD
ARGUMENT	ADP	—	—	CASE	**SWAP**
ARGUMENT	AUX	QUES	—	AUX:Q	**SWAP**
ARGUMENT	VERB	NEG	—	COP:NEG	**SWAP**
CONJUNCTION	—	—	*(de | ki | mi)*	MARK	**CLAUSAL**
CONJUNCTION	—	—	**NOT** *(de | ki | mi)*	CC	**CLAUSAL**
COORDINATION	—	—	—	CONJ	**CLAUSAL**
INTENSIFIER	NUM	—	*ise*	DISCOURSE	**CLAUSAL**
PUNCTUATION	SMILEY	—	—	DISCOURSE	**CLAUSAL**
PUNCTUATION	—	—	—	PUNCT	**CLAUSAL**
VOCATIVE	INTJ | SYM	—	—	DISCOURSE	**CLAUSAL**
VOCATIVE	NOUN | PROPN	—	—	VOCATIVE	**CLAUSAL**

Table 8: Dependency mapping: Other dependencies, adjusting the typology.

For any remaining tokens whose dependencies were not updated by any of the given mapping rules, a catch-all UD relation was assigned according to its converted part of speech. Tokens with the part-of-speech tags ADP, CONJ, INTJ and PUNCT were respectively attached the dependency relations CASE, CC, VOCATIVE and PUNCT. Those with the tags ADJ, ADV, DET and NUM were respectively given the AMOD, ADVMOD, DET and NUMMOD relations. Any other token was assigned the NMOD relation.

2.5 Postprocessing

After the adjustments to segmentation and the conversion of part-of-speech tags, morphological features and dependency relations, we applied postprocessing routines to each sentence to ensure they constitute valid dependency trees. This step was also necessary in order to circumvent some cases in the original IMST Treebank where sentences did not have a unique token with the sentence root as the head. These cases were often due to dependencies such as CONJUNCTION, PUNCTUATION and VOCATIVE, which depended on the sentence root in certain contexts as required by the dependency grammar. Otherwise, a small number of annotation errors which broke the unique root constraint were also present in the original treebank, and these warranted addressing as well.

Initially, every token depending on the sentence root with a non-ROOT dependency was reassigned the clausal head (or, if not applicable, the sentential head) as its new head. The remaining sentences that still broke the constraint were artifacts of annotation errors. For these sentences, an additional treeification procedure was applied to break all cycles and ensure the possibility of reaching the root from any token.

For sentences with no rooted token (and at least one obligatory cycle), the rightmost token that was part of a cycle was considered the sentential head and connected to the sentence root with the dependency relation ROOT. For any other cycles, the token with the most dependents in the cycle was considered a clausal head and connected to the sentential head, keeping its original dependency relation. Finally, if a sentence had multiple rooted tokens, the rightmost rooted token with a VERB category (or, in the absence of rooted VERB tokens, simply the rightmost rooted token) was considered the sentential head, and the other rooted tokens were connected to that token with the dependency relation CONJ.

3 The IMST Treebank

The IMST Treebank is a Turkish dependency treebank of well-edited sentences from a wide range of domains, fully annotated for morphological analyses and dependency relations. The treebank underwent substantial changes since its unofficial conception in 2014 and was at version 1.3 when it was officially released.[4]

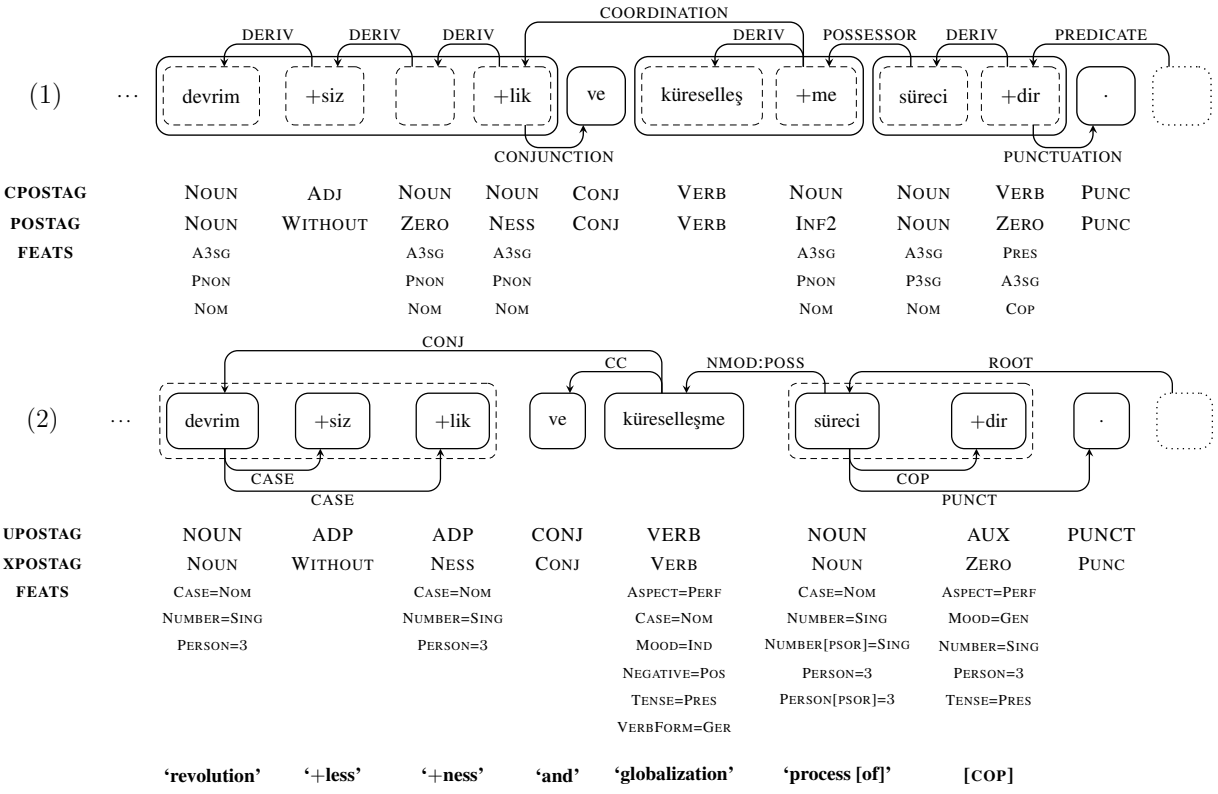

Figure 1: An example of a partial sentence, "*...devrimsizlik ve küreselleşme sürecidir.*" ("**...is the process of revolutionlessness and globalization.**"), before (1) and after (2) the conversion, extracted from the IMST and IMST-UD treebanks.

The IMST Treebank was annotated using its own annotation framework, which is based on that of the METU-Sabancı Treebank and radically different from the UD framework. Figure 1 compares a partial sentence from the IMST Treebank before and after the conversion. The + sign is used for convenience as a suffix marker, and does not actually occur in the treebank. The token enclosures denote either IG sets (in the original treebank sentences) or multi-word tokens (in converted sentences). As shown in the example, these multi-word groups were converted to a head-first typology, whereas coordination structures remained head-final. This is because the final token in a coordination structure always retains all inflection, whereas suffixes shared by all the tokens may be dropped in the others.

Table 9 presents a selection of comparative statistics, including the total numbers of sentences, tokens and dependency counts as well as the counts of unique part-of-speech tags, morphological features and dependency relations for the baseline and converted versions of the IMST Treebank, as a preamble to the parsing tests described in Section 4. We use the treebank's version 1.3.1 as the baseline for the UD conversion. For this reason, the statistics provided in this section are slightly different from those given in the IMST Treebank's original publication (Sulubacak et al., 2016).

[4]The latest version of the treebank is available for research purposes on `http://tools.nlp.itu.edu.tr`

	IMST	IMST–UD
# *Sentences*	5635	5635
# *(Orthographic) Tokens*	56423	56423
# *(Syntactic) Words*	63072	58085
# *Dependencies*	56423 *(excl.* DERIV*)*	58085
	63072 *(incl.* DERIV*)*	
# *Projective Dependencies*	61849	55043
# *Non-projective Dependencies*	1223	3042
# *(Unique) Parts of Speech*	11	14
# *(Unique) Morphological Features*	47	67
# *(Unique) Dependency Relations*	16	29

Table 9: Comparative statistics for the IMST Treebank and the IMST-UD Treebank.

4 Evaluation

In this section, we present our statistical analysis on the parsing performances of the original and converted versions of the IMST Treebank.

4.1 Preliminaries

For our parsing tests, we employ the same MaltParser (Nivre et al., 2007) configuration as in many previous studies on the METU-Sabancı Treebank (Eryiğit, 2006; Eryiğit et al., 2008; Eryiğit et al., 2011; Sulubacak and Eryiğit, 2013) and the IMST Treebank (Sulubacak et al., 2016). In compliance with the parsing procedures used in the cited studies, we eliminate non-projective sentences from each training set, as this practice was shown to boost overall performance[5] (Eryiğit et al., 2008; Eryiğit et al., 2011).

In further accordance with the cited studies, we use the conventional labeled and unlabeled attachment scores as our evaluation metrics. Although both scores are essentially based on the ratio of correct predictions to all tokens, they differ in which predictions they accept as correct. While a correct prediction of the head token suffices for the unlabeled attachment score (UAS), the labeled attachment score (LAS) also requires the dependency relation to be correctly predicted. Furthermore, dependencies with the relation DERIV[6] are excluded from evaluation for the baseline version, as they are considered trivial.

4.2 Parsing Scores

The parsing scores given in Table 10 were calculated via ten-fold cross-validation on the baseline *(left)* and the UD *(right)* versions of the IMST Treebank. A comparison of the scores before and after the conversion to UD shows that there has been a noticeable improvement in the labeled attachment score, despite the consequential increase in the number of unique POS tags, morphological features and dependency labels, as previously shown in Table 9. However, there has been no apparent progress in the unlabeled attachment score. Since head indices had also been adjusted as part of the mapping procedure, the similarity in the scores is likely a favorable coincidence. Considering both scores, it is evident that the UD framework has been more accommodating for the IMST Treebank over the current parsing setup.

	IMST	IMST–UD
LAS	$75.4 \pm 0.2\%$	$77.1 \pm 0.2\%$
UAS	$83.8 \pm 0.3\%$	$83.8 \pm 0.2\%$

Table 10: Attachment scores.

[5] We tested this on the UD version of the IMST Treebank as well, and including non-projective sentences in training indeed caused a drop of 2.9 percentage points in the average labeled attachment score compared to training without non-projective sentences.

[6] The DERIV relation is used in the annotation framework of the original IMST Treebank to mark intra-token dependencies between morphosyntactic units. Each such unit depends on the next, and the rightmost unit is considered to be the head.

5 Conclusion

In this paper, we described our procedure for converting the morphological and syntactic tagset of the IMST Treebank to comply with the UD standard. In doing so, we presented a specific application of the UD guidelines to the annotation of parts of speech, morphological features and dependency relations in Turkish. We also introduced the IMST-UD Treebank, which was automatically converted from the IMST Treebank and became the first Turkish treebank to be in a UD release. We also evaluated the parsing performances on the IMST and IMST-UD treebanks and found that there is a noticeable improvement in parsing performances after conversion, which suggests that the UD framework is at least as viable for Turkish as the original annotation framework of the IMST Treebank.

Acknowledgements

We would like to thank Birsel Karakoç, Hüner Kaşıkara and Tuğba Pamay for their valuable insights and discussions, and the Uppsala University for kindly hosting a meeting for us during the initial stages of our effort. We would also like to offer our gratitude to our anonymous reviewers for meticulously proofreading our manuscript.

References

Nart B. Atalay, Kemal Oflazer, and Bilge Say. 2003. The annotation process in the Turkish Treebank.

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X Shared Task on multilingual dependency parsing. In *Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL)*, pages 149–164. Association for Computational Linguistics.

Çağrı Çöltekin. 2015. A grammar-book treebank of Turkish. In Markus Dickinson, Erhard Hinrichs, Agnieszka Patejuk, and Adam Przepiórkowski, editors, *Proceedings of the 14th workshop on Treebanks and Linguistic Theories (TLT 14)*, pages 35–49.

Çağrı Çöltekin. 2016. (when) do we need inflectional groups? In *Proceedings of The 1st International Conference on Turkic Computational Linguistics*, page (to appear).

Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of 5th International Conference on Language Resources and Evaluation (LREC)*, volume 6, pages 449–454.

Marie-Catherine De Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal Stanford Dependencies: a cross-linguistic typology. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*, Reykjavík, Iceland, May. European Language Resources Association (ELRA).

Gülşen Eryiğit and Kemal Oflazer. 2006. Statistical dependency parsing of Turkish. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 89–96, Trento, April.

Gülşen Eryiğit. 2006. *Dependency Parsing of Turkish*. Ph.D. thesis, Istanbul Technical University.

Gülşen Eryiğit, Joakim Nivre, and Kemal Oflazer. 2008. Dependency parsing of Turkish. *Computational Linguistics*, 34(3):357–389.

Gülşen Eryiğit, Tugay Ilbay, and Ozan Arkan Can. 2011. Multiword expressions in statistical dependency parsing. In *Proceedings of the 2nd Workshop on Statistical Parsing of Morphologically Rich Languages (SPMRL)*, Dublin, Ireland, October. Association for Computational Linguistics.

Dilek Zeynep Hakkani-Tür, Kemal Oflazer, and Gökhan Tür. 2002. Statistical morphological disambiguation for agglutinative languages. *Computers and the Humanities*, 36(4):381–410.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülşen Eryiğit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13:95–135, 6.

Kemal Oflazer, Bilge Say, Dilek Zeynep Hakkani-Tür, and Gökhan Tür. 2003. Building a Turkish treebank. In Anne Abeille, editor, *Building and Exploiting Syntactically-Annotated Corpora*. Kluwer Academic Publishers.

Kemal Oflazer. 1999. Dependency parsing with an extended finite state approach. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 254–260, College Park, Maryland, USA. Association for Computational Linguistics.

Kemal Oflazer. 2003. Dependency parsing with an extended finite-state approach. *Computational Linguistics*, 29(4):515–544.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*, pages 2089–2096.

Umut Sulubacak and Gülşen Eryiğit. 2013. Representation of morphosyntactic units and coordination structures in the Turkish dependency treebank. In *Proceedings of the 4th Workshop on Statistical Parsing of Morphologically-Rich Languages (SPMRL)*, pages 129–134, Seattle, Washington, USA, October. Association for Computational Linguistics.

Umut Sulubacak, Tuğba Pamay, and Gülşen Eryiğit. 2016. IMST: A revisited Turkish dependency treebank. In *Proceedings of the 1st International Conference on Turkic Computational Linguistics (TurCLing)*, Konya, Turkey, 3 April.

Reut Tsarfaty. 2013. A unified morpho-syntactic scheme of Stanford Dependencies. In *Proceedings of 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 578–584.

Daniel Zeman. 2008. Reusable tagset conversion using tagset drivers. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC)*, pages 213–218.